

UNITED STATES PATENT APPLICATION

of

Jon Avner

and

Soner Terek

for

METHODS AND SYSTEMS FOR ALLOWING THIRD PARTY CLIENT APPLICATIONS TO INFLUENCE IMPLEMENTATION OF HIGH-LEVEL DOCUMENT COMMANDS

WORKMAN, NYDEGGER & SEELEY
A PROFESSIONAL CORPORATION
ATTORNEYS AT LAW
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UTAH 84111

[illegible]

1
2
3
4
5
6
7
8
9
0
1
2
3
4
5
6
7
8
9
0
1
2
3
4

2

3
4
5
6

8

9
0
1
2
3
4

5
6
7
8

19
20
21
22
23
24

1 associated with given files, or folders. For each document, there is typically more than one
2 table that may associated with the document by, for example, describing properties of the
3 document.

4 A document (e.g., document 1, 2, 3 and 4 in Figure 1) is an identifiable entity from
5 the viewpoint of applications that use the database management system. For example, a
6 folder or an item such as an electronic mail message within the folder may be considered to
7 be a document since they are identified as entities from the viewpoint of the application
8 that uses the database management system. For example, in the context of electronic
9 messaging, an application may present folders such as "in-boxes" and "out-boxes" to a
10 user with corresponding electronic mail messages in the folders. These folders and
11 electronic mail messages are "documents" from the viewpoint of the application.
12 Similarly, files and directories within a file system may also be documents from the
13 viewpoint of the application that uses the database management system. The tables within
14 the underlying database are not considered to be documents since they are identified
15 internal to the database management system, and not at the higher level of the application
16 that uses the database management system.

17 The database application 120 generates high-level document commands (e.g., high-
18 level command 121) that relate to operations to be performed on a document. Examples of
19 such document commands might include operations such as, for example, move folder,
20 move message, delete message, copy folder, copy file, and so forth.

21 Each of these high-level document commands is received by the database engine
22 130 which implements the high-level document commands by executing a number of table
23 level commands that result in database tables being updated. Typically, the database
24 engine would use the disk access module 140 of an operating system to produce the

1 physical control signals necessary to read and write the appropriate sectors in the disk,
2 each table comprising one or more possibly discontinuous sectors on the disk.

3 Once the high-level document command is implemented, the database management
4 system may notify other client applications using notification module 150 that the high-
5 level document command was implemented. Then, the other client applications may
6 perform a variety of processes in response to that notification. For example, if a new
7 electronic mail messages is added into a public folder, the other client applications that
8 have subscribed to that public folder may receive notification of the new electronic mail
9 message so that they can update there local database to reflect the change, or perhaps
10 refresh the screen if it is currently displaying the contents of the public folder.

11 In the conventional method, the notification is only dispatched after the high-level
12 document command is implemented. There is nothing the client application can do to
13 affect how the high-level document command is implemented, what additional actions are
14 to be taken in the database at the same time the high-level document command is
15 implemented, or whether the high-level document command is to be implemented at all.
16 Therefore, what are desired are methods and systems for allowing more flexibility for a
17 third party client application to affect how a high-level document command is
18 implemented in a database.

SUMMARY OF THE INVENTION

The principles of the present invention provide for a way for third party client applications to influence how a high-level document command is implemented by a database management system. The database management system is issued a number of high level-document commands which are intended to be implemented in the underlying database. Such high-level document commands include high-level commands that relate to actions to be taken on documents such as folders, files, messages and other entities that are identified at the level of the application that uses the database management system. For example, in electronic messaging applications, one high-level document command may be to add a mail message to a folder.

Unlike conventional database management systems, however, the database management system in accordance with the invention temporarily foregoes implementing the high-level document command in the database. Instead, the database management system identifies any third party client applications that are to be notified when the high-level document command is received. Then, the database management system dispatches a notification. The client application is thus made aware of the notification so that the client application may return back implementation instructions on how to implement the high-level document command. The database management system receives the implementation instructions and follows the instructions in implementing the high-level document command.

For example, the instruction may be for preventing the implementation of the high-level document command altogether. The instruction may also be to change how the high-level document command is implemented. The instruction may even be for implementing one or more high-level document commands in addition to the received high-level

1 document command. Thus, the notified third party client application is given extensive
2 control over how the high-level document command is implemented.

3 Additional features and advantages of the invention will be set forth in the
4 description which follows, and in part will be obvious from the description, or may be
5 learned by the practice of the invention. The features and advantages of the invention may
6 be realized and obtained by means of the instruments and combinations particularly
7 pointed out in the appended claims. These and other features of the present invention will
8 become more fully apparent from the following description and appended claims, or may
9 be learned by the practice of the invention as set forth hereinafter.

BRIEF DESCRIPTION OF THE DRAWINGS

In order to describe the manner in which the above-recited and other advantages and features of the invention can be obtained, a more particular description of the invention briefly described above will be rendered by reference to specific embodiments thereof which are illustrated in the appended drawings. Understanding that these drawings depict only typical embodiments of the invention and are not therefore to be considered to be limiting of its scope, the invention will be described and explained with additional specificity and detail through the use of the accompanying drawings in which:

Figure 1 illustrates a hierarchical view of a database management system in accordance with the prior art;

Figure 2 illustrates an exemplary system that provides a suitable operating environment for the present invention;

Figure 3 illustrates a hierarchical view of a database management system in accordance with the present invention; and

Figure 4 illustrates a flowchart of a method for allowing a third party client application to influence how a high-level document command is implemented in the database of Figure 3.

DETAILED DESCRIPTION OF THE INVENTION

The present invention extends to both methods and systems for allowing a third party client application to influence how a high-level document command is implemented in a database. Appropriate third party client applications are notified when particular high-level document commands are received in the database management system. However, before implementing the high-level document command in the database, the database management system awaits for any instructions from the appropriate third party application on how the high-level document command is to be implemented.

If there are no such intervening instructions, the high-level document command is implemented in the database. If there are such intervening instructions, these intervening instructions from the third party client applications are considered in how the high-level document command is implemented. For example, the intervening instruction may be to prevent the implementation of the high-level document command entirely. It may also be to alter the way that the high-level document command is implemented. A third party client application may also have the database management system perform high-level document commands in addition to the original high-level document command. When performing additional high-level operations, all of the high-level operations may be performed atomically in a single group operation so that either all of the high-level operations in the group operation are performed, or none are performed at all.

The embodiments of the present invention may comprise a special purpose or general purpose computer including various computer hardware, as discussed in greater detail below. Embodiments within the scope of the present invention also include computer-readable media for carrying or having computer-executable instructions or data structures stored thereon. Such computer-readable media can be any available media

1 which can be accessed by a general purpose or special purpose computer. By way of
2 example, and not limitation, such computer-readable media can comprise physical storage
3 media such as RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic
4 disk storage or other magnetic storage devices, or any other medium which can be used to
5 carry or store desired program code means in the form of computer-executable instructions
6 or data structures and which can be accessed by a general purpose or special purpose
7 computer.

8 When information is transferred or provided over a network or another
9 communications connection (either hardwired, wireless, or a combination of hardwired or
10 wireless) to a computer, the computer properly views the connection as a computer-
11 readable medium. Thus, any such a connection is properly termed a computer-readable
12 medium. Combinations of the above should also be included within the scope of
13 computer-readable media. Computer-executable instructions comprise, for example,
14 instructions and data which cause a general purpose computer, special purpose computer,
15 or special purpose processing device to perform a certain function or group of functions.

16 Figure 2 and the following discussion are intended to provide a brief, general
17 description of a suitable computing environment in which the invention may be
18 implemented. Although not required, the invention will be described in the general context
19 of computer-executable instructions, such as program modules, being executed by
20 computers in network environments. Generally, program modules include routines,
21 programs, objects, components, data structures, etc. that perform particular tasks or
22 implement particular abstract data types. Computer-executable instructions, associated
23 data structures, and program modules represent examples of the program code means for
24 executing steps of the methods disclosed herein. The particular sequence of such

1 executable instructions or associated data structures represent examples of corresponding
2 acts for implementing the functions described in such steps.

3 Those skilled in the art will appreciate that the invention may be practiced in
4 network computing environments with many types of computer system configurations,
5 including personal computers, hand-held devices, multi-processor systems,
6 microprocessor-based or programmable consumer electronics, network PCs,
7 minicomputers, mainframe computers, and the like. The invention may also be practiced
8 in distributed computing environments where tasks are performed by local and remote
9 processing devices that are linked (either by hardwired links, wireless links, or by a
10 combination of hardwired or wireless links) through a communications network. In a
11 distributed computing environment, program modules may be located in both local and
12 remote memory storage devices.

13 With reference to Figure 2, an exemplary system for implementing the invention
14 includes a general purpose computing device in the form of a conventional computer 220,
15 including a processing unit 221, a system memory 222, and a system bus 223 that couples
16 various system components including the system memory 222 to the processing unit 221.
17 The system bus 223 may be any of several types of bus structures including a memory bus
18 or memory controller, a peripheral bus, and a local bus using any of a variety of bus
19 architectures. The system memory includes read only memory (ROM) 224 and random
20 access memory (RAM) 225. A basic input/output system (BIOS) 226, containing the basic
21 routines that help transfer information between elements within the computer 220, such as
22 during start-up, may be stored in ROM 224.

23 The computer 220 may also include a magnetic hard disk drive 227 for reading
24 from and writing to a magnetic hard disk 239, a magnetic disk drive 228 for reading from

1 or writing to a removable magnetic disk 229, and an optical disk drive 230 for reading
2 from or writing to removable optical disk 231 such as a CD-ROM or other optical media.
3 The magnetic hard disk drive 227, magnetic disk drive 228, and optical disk drive 230 are
4 connected to the system bus 223 by a hard disk drive interface 232, a magnetic disk drive-
5 interface 233, and an optical drive interface 234, respectively. The drives and their
6 associated computer-readable media provide nonvolatile storage of computer-executable
7 instructions, data structures, program modules and other data for the computer 220.
8 Although the exemplary environment described herein employs a magnetic hard disk 239,
9 a removable magnetic disk 229 and a removable optical disk 231, other types of computer
10 readable media for storing data can be used, including magnetic cassettes, flash memory
11 cards, digital video disks, Bernoulli cartridges, RAMs, ROMs, and the like.

12 Program code means comprising one or more program modules may be stored on
13 the hard disk 239, magnetic disk 229, optical disk 231, ROM 224 or RAM 225, including
14 an operating system 235, one or more application programs 236, other program modules
15 237, and program data 238. A user may enter commands and information into the
16 computer 220 through keyboard 240, pointing device 242, or other input devices (not
17 shown), such as a microphone, joy stick, game pad, satellite dish, scanner, or the like.
18 These and other input devices are often connected to the processing unit 221 through a
19 serial port interface 246 coupled to system bus 223. Alternatively, the input devices may
20 be connected by other interfaces, such as a parallel port, a game port or a universal serial
21 bus (USB). A monitor 247 or another display device is also connected to system bus 223
22 via an interface, such as video adapter 248. In addition to the monitor, personal computers
23 typically include other peripheral output devices (not shown), such as speakers and
24 printers.

1 The computer 220 may operate in a networked environment using logical
2 connections to one or more remote computers, such as remote computers 249a and 249b.
3 Remote computers 249a and 249b may each be another personal computer, a server, a
4 router, a network PC, a peer device or other common network node, and typically include
5 many or all of the elements described above relative to the computer 220, although only
6 memory storage devices 250a and 250b and their associated application programs 236a and
7 236b have been illustrated in Figure 2. The logical connections depicted in Figure 2
8 include a local area network (LAN) 251 and a wide area network (WAN) 252 that are
9 presented here by way of example and not limitation. Such networking environments are
10 commonplace in office-wide or enterprise-wide computer networks, intranets and the
11 Internet.

12 When used in a LAN networking environment, the computer 220 is connected to
13 the local network 251 through a network interface or adapter 253. When used in a WAN
14 networking environment, the computer 220 may include a modem 254, a wireless link, or
15 other means for establishing communications over the wide area network 252, such as the
16 Internet. The modem 254, which may be internal or external, is connected to the system
17 bus 223 via the serial port interface 246. In a networked environment, program modules
18 depicted relative to the computer 220, or portions thereof, may be stored in the remote
19 memory storage device. It will be appreciated that the network connections shown are
20 exemplary and other means of establishing communications over wide area network 252
21 may be used.

22 Figure 3 illustrates a database management system 300 in accordance with the
23 present invention. Although, not required, the database management system 300 may be
24 implemented in the computing environment shown in Figure 2. The database application

1 320 issues high-level document commands such as command 321. The document
2 command may be, for example, to move, update, copy, add, or delete the document. For
3 each high-level document command, the database management system 300 notifies, via
4 notification module 350, the appropriate client applications(s).

5 However, in contrast to the prior art database management system, the database
6 management system 300 includes a means for allowing one or more client applications to
7 affect how the received high-level document is to be implemented, if at all, in the database.
8 This means is illustrates in Figure 3 as the instruction receiver module 360 which may be
9 implemented by software, hardware, or a combination of software and hardware. The
10 instruction receiver module may be implemented separately or perhaps by the database
11 engine 330, the database application 320, another component of the database management
12 system, or a combination thereof.

13 Figure 4 illustrates a flowchart of a method 400 for allowing third party
14 applications to influence the implementation of a high-level document commands in a
15 database. Acts implemented by the database management system are listed under the
16 column entitled "DATABASE MANAGEMENT SYSTEM" while those act performed by
17 the client application are listed under the column entitled "CLIENT APPLICATION."

18 The method is initiated by the database management system receiving a high-level
19 document command (act 401). The command is high-level in the sense that the command
20 is for an operation to be performed on a document. The document is an entity such as an
21 electronic mail message or a folder, which is identifiable at the high-level of the client
22 application that is using the database management system. Such high-level applications
23 typically have no knowledge of the tables in the database or how such tables would be
24

1 altered to implement the high-level document command. Thus, tables in databases are not
2 considered to be "documents" in this description and in the claims.

3 After the database management system receives the high-level command, but
4 before implementing the high-level command, the database management system allows
5 third party client applications to affect how the high-level document command is
6 implemented in the database. Accordingly, embodiments within the scope of the present
7 invention include the database management system implementing a step for allowing one
8 or more client applications to affect how the high-level document command is
9 implemented, if at all (step 402). In the example shown in Figure 4, this step includes acts
10 403, 404, 407 and 408.

11 For the received high-level document command, the database management system
12 identifies any client applications that are to be notified when the database management
13 system receives the high-level document command (act 403). These client applications
14 may be identified by being client applications that are notified by default when the
15 database management system receives a high-level document command meeting specified
16 criteria. Alternatively or in addition, client applications may be identified by having a
17 subscription for notifications to occur when the database management system receives
18 high-level document commands meeting specified criteria.

19 Once identified, the client applications are notified of the receipt of the high-level
20 document command (act 404). This may be performed via messaging over a network if the
21 database management system and the client application are located on different machines.
22 Also, this may be performed by the notification being passed through a function call as
23 when the database management system and the client application are located on the same
24 machine.

1 The client application then receives the notification (act 405) and then dispatches
2 instructions back to the database management system (act 406) on how to implement the
3 high-level document command. For example, the client application may instruct the
4 prevention of the implementation altogether, or perhaps how the high-level document
5 command is implemented, or perhaps even what additional high-level document
6 commands are to perform in addition to the received high-level document command. The
7 database management system then receives these implementation instructions (act 407) and
8 then follows the instructions when actually implementing the high-level document
9 command (act 408).

10 The implementation instruction may be prevented from being implemented
11 altogether if so instructed. For example, suppose that a client application is to be notified
12 when a high-level document command is received that is for adding a new electronic mail
13 message into an electronic inbox, the electronic mail message containing certain offensive
14 words. Now suppose that such a high-level document command is, in fact, received at the
15 database management system. The client application will thus be notified. The client
16 application may dispatch instructions to prevent the high-level document command from
17 being implemented at all. The user may have configured the client application to so
18 instruct the database management system to prevent implementation in this case so as to
19 avoid having to encounter offensive words.

20 The implementation instruction may be for changing the way that the high-level
21 document command is implemented. For example, instead of preventing the addition of
22 electronic mail messages that contain certain offensive words, the user may have
23 configured the client application to instruct the database management system to type over
24

1 the offensive words with random alphanumeric text before adding the electronic mail
2 message into the inbox.

3 The implementation instruction may also be to perform one or more high-level
4 document commands in addition to the received high-level document command. For
5 example, the user may configure the client application to instruct the database management
6 system so that any electronic mail message being marked as "urgent" are not only added to
7 the user's inbox, but also are copied to the in-box of the user's staff, and the in-box of the
8 user's supervisor as well.

9 Sub A7 When performing additional high-level document commands over and above the
10 received high-level document commands, the document commands may be implemented
11 as a "group operation" in which all of the document commands are implemented in a
12 single transaction, all of the high-level document command being implemented, or none at
13 all. Group operations are described in co-pending United States application serial number
14 [NOT YET ASSIGNED - ATTORNEY DOCKET NUMBER 13768.171], filed on the
15 same date herewith, and entitled "METHODS AND SYSTEMS FOR PERFORMING
16 HIGH-LEVEL GROUP OPERATIONS IN A DATABASE MANAGEMENT SYSTEM",
17 which is incorporated herein by reference in its entirety.

18 As described above, the principles of the present invention allow for third party
19 client applications to influence how a high-level document command is to be implemented.
20 This give more control to client applications that desire to tailor how certain high-level
21 document commands are to be implemented.

22 The present invention may be embodied in other specific forms without departing
23 from its spirit or essential characteristics. The described embodiments are to be considered
24 in all respects only as illustrative and not restrictive. The scope of the invention is,

1 therefore, indicated by the appended claims rather than by the foregoing description. All
2 changes which come within the meaning and range of equivalency of the claims are to be
3 embraced within their scope.

4 What is claimed and desired to be secured by United States Letters Patent is:
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24